



University of
Pittsburgh

Algorithms and Data Structures 2

CS 1501

Fall 2021

Sherif Khattab

ksm73@pitt.edu

(Slides are adapted from Dr. Ramirez's and Dr. Farnan's CS1501 slides.)

Announcements

- Upcoming Deadlines
 - Homework 1: today at 11:59 pm
 - Homework 2: Monday 1/31 at 11:59 pm
 - Lab 1: Friday 1/28 at 11:59 pm
- Assignment 1 not yet posted (sorry about that)
- CourseMIRROR consent form and pre-survey

Previous lecture

- Backtracking solution for the Boggle Game problem
- General backtracking template

CourseMIRROR Reflections

Searching Problem

- Input:
 - a (large) dynamic set of data items in the form of
 - (key, value) pairs
 - What does dynamic mean?
 - a target *key* to search for
- Output:
 - if *key* exists in the set: return the corresponding value
 - otherwise, return *key not found*

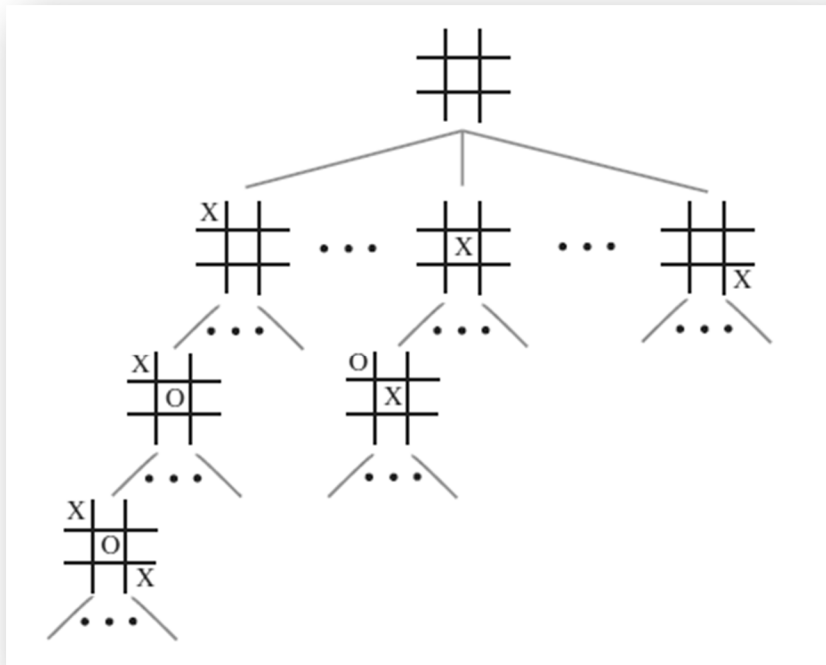
Let's create an ADT!

- The Symbol Table ADT
 - A set of (key, value) pairs
- Operations of the ST ADT
 - insert
 - search
 - delete

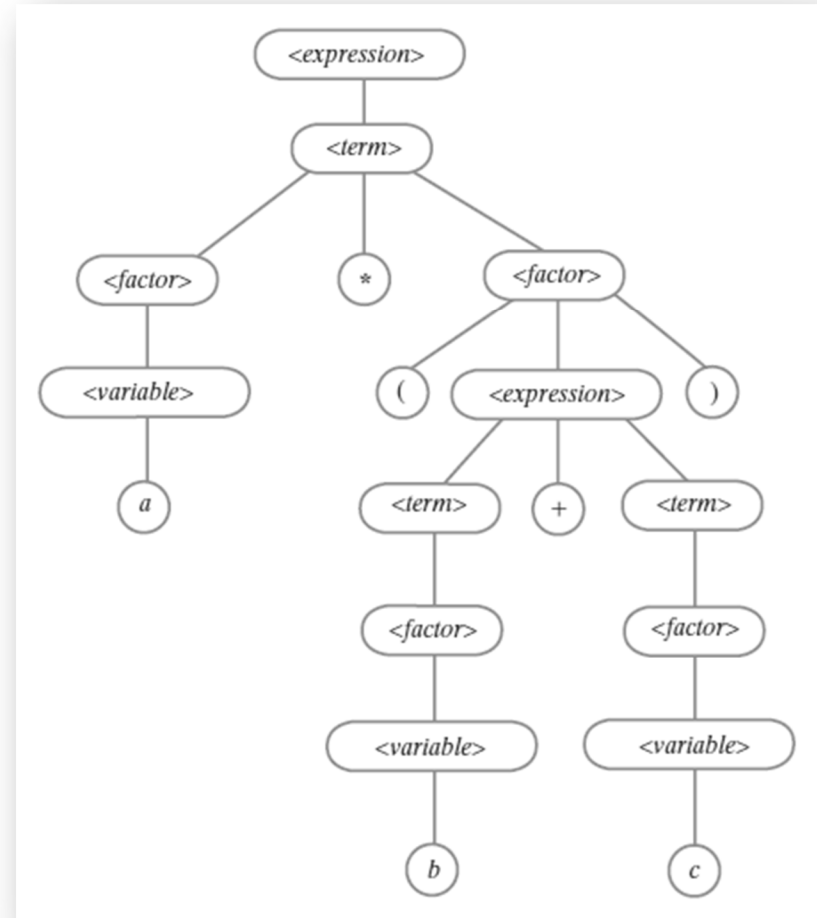
Symbol Table Implementations

- Array
 - Unsorted
 - Sorted
- Linked List
 - Unsorted
 - Sorted
- What if we use a non-linear data structure?
 - a Tree?

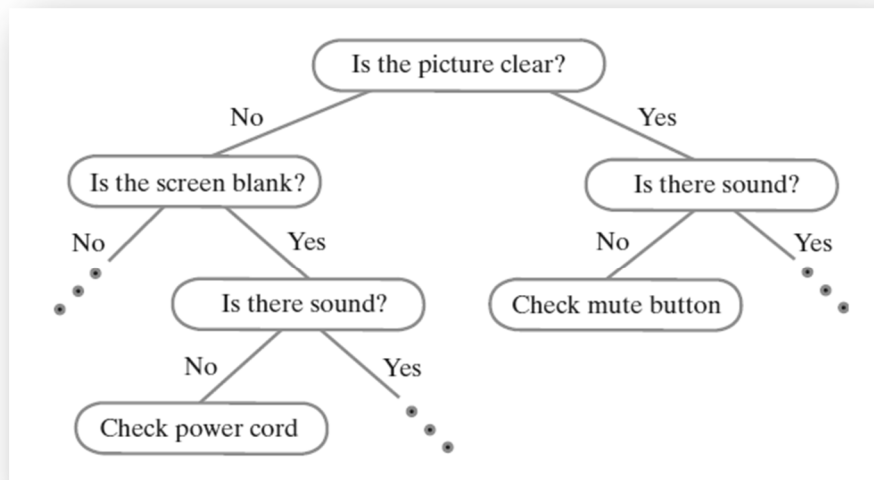
Examples of Trees



Game Tree

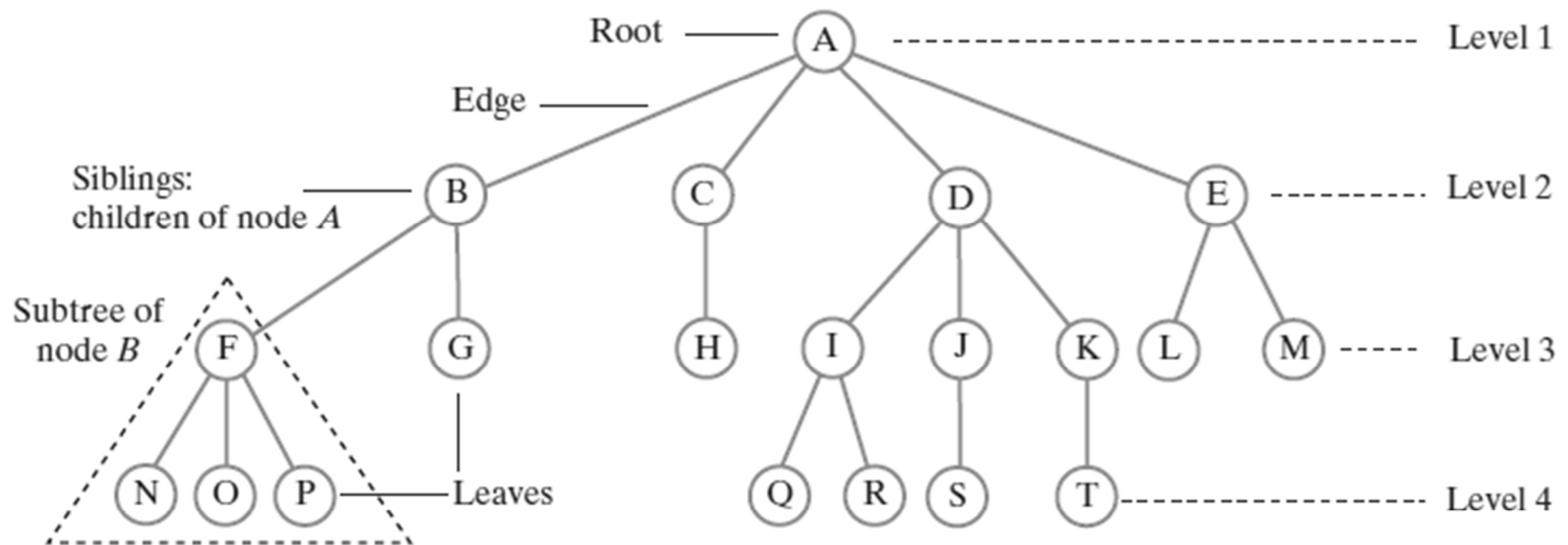


Parse Tree



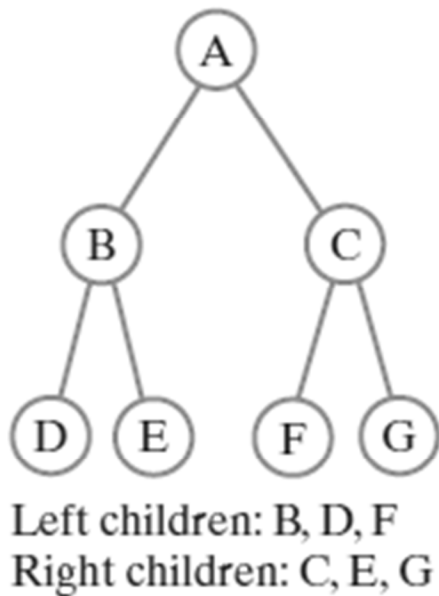
Decision Tree

Tree Terminology

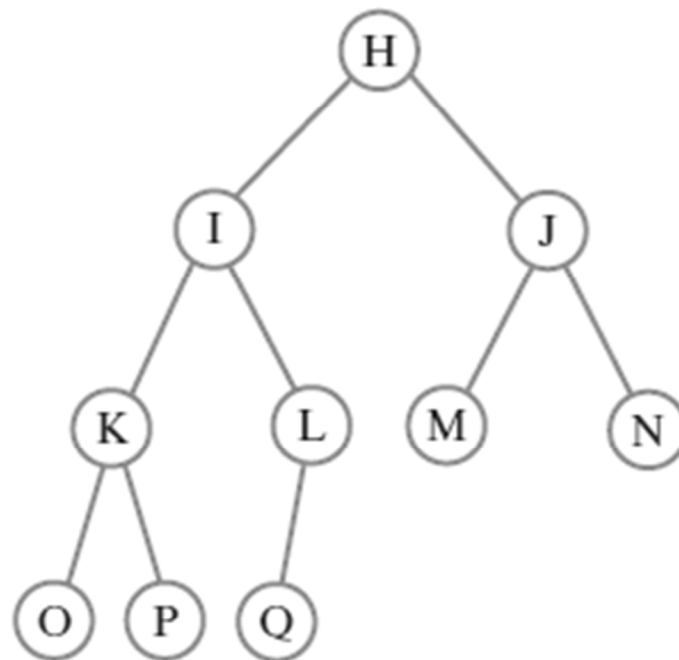


Binary Trees

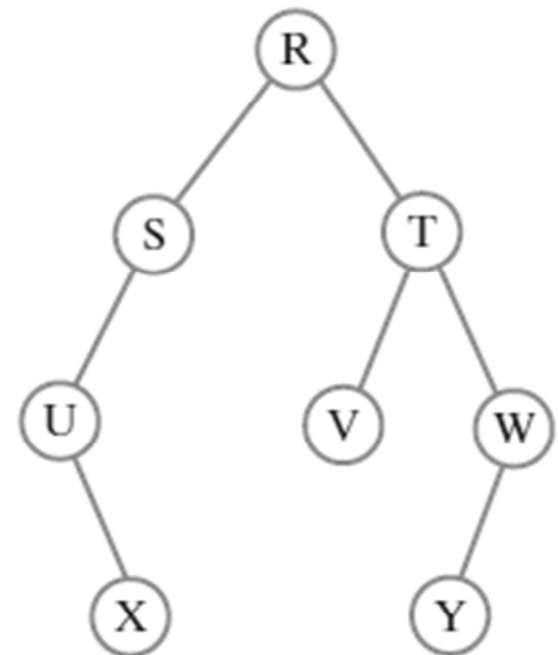
(a) Full tree



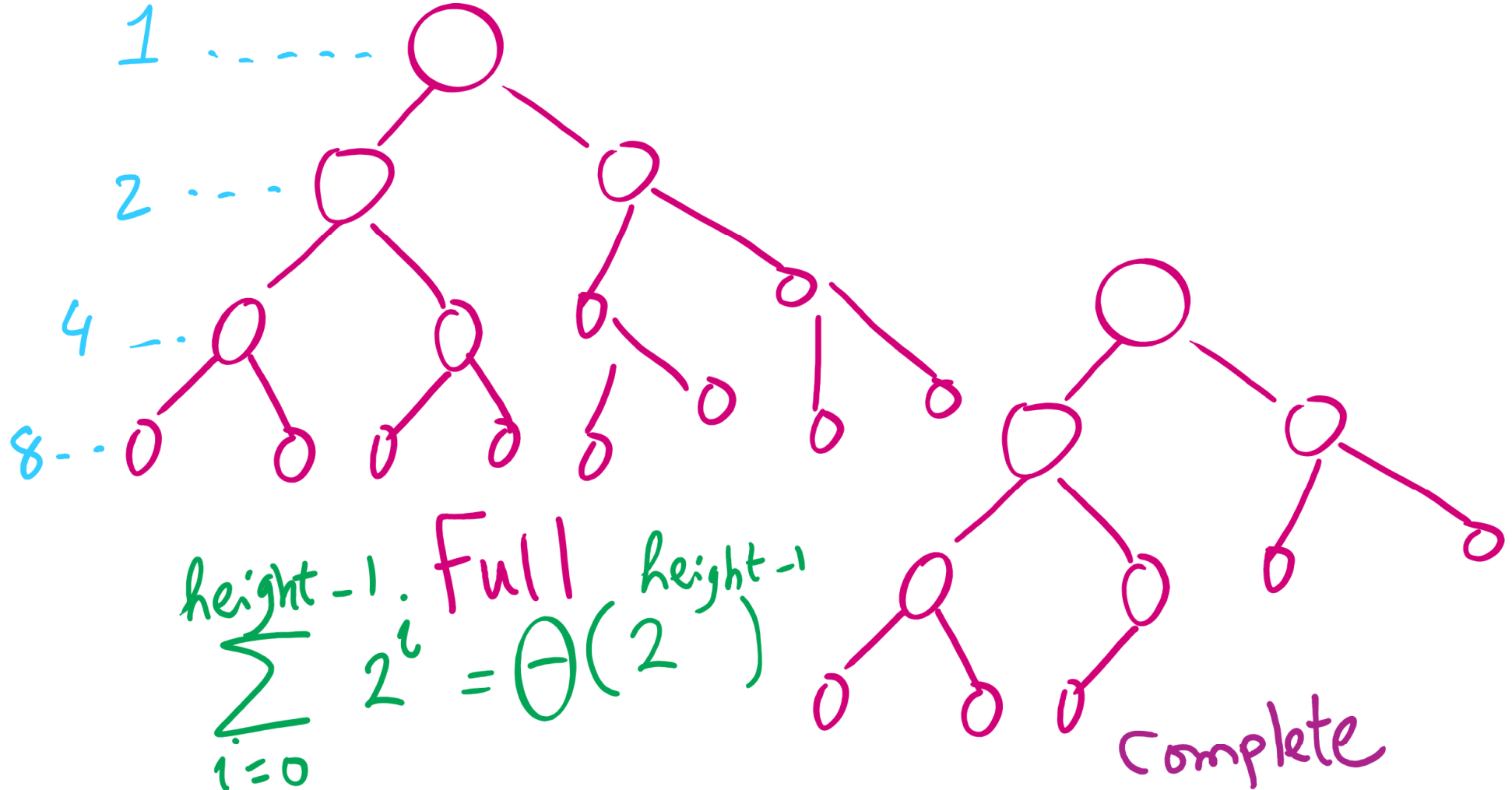
(b) Complete tree



(c) Tree that is not full and not complete



Full vs. Complete Tree



Tree Search Take 1

- *Traverse* every node of the tree
 - Is the key inside the node equal to the target *key*?
- How can we traverse the tree?

Tree Search Take 1

What is the runtime?

Can we do better?

Can we traverse the tree more intelligently?

Tree Search Take 2: Binary Search Tree

- Search Tree Property
 - $\text{left} < \text{root} < \text{right}$

Please submit your reflections by using the CourseMIRROR App

If you are having a problem with CourseMIRROR, please send an email to coursemirror.development@gmail.com

8/29/2022